REMARKS

Claims 1, 4, and 7-24 are currently pending in the subject application and are presently under consideration.

Applicants' representative thanks the Examiner for considering the remarks tendered by telephone on 15 August 2008 at 3:30 p.m. PST. It is hoped that the remarks will facilitate a more meaningful dialog during the prosecution of the pending application.

Favorable reconsideration of the subject patent application is respectfully requested in view of the comments and amendments herein.

I. Rejection of Claims 10 and 14 Under 35 U.S.C §112

Claims 10 and 14 stand rejected under 35 U.S.C §112, first paragraph, as failing to comply with the enablement requirement. Applicants' representative disagrees with the Examiner's position for at least the following reasons. In regard to claim 10, the specification states that incoming viruses can be manipulated before "reaching the computer" such that, for example, a packed executable can be unpacked and inspected prior to being able to infect a computer. One of elementary skill in the art at the time of the invention would clearly understand that the incoming data is "at the computer" (*e.g.*, it is resident in memory somewhere in relation to the computer) but that the incoming code has not "reached the computer" because the code is logically isolated to prevent it from executing without first being inspected. Further, one of ordinary skill in the art would have easily understood that this can be achieved by placing the incoming code into a memory and further more it would have been well known at the time of the invention that such a memory could be a RAM module (*e.g.*, most *circa* 2004 computers employed RAM as a memory component). Moreover, this RAM was well known to be logically partitionable into safe memory areas. Similarly, hard drives were well known and these were also well known for being partitionable into safe partitions. Applicants' representative contends that one of even the most basic skill in the art at the time of the invention would regard RAM, quarantine areas and logical partitions as well known in the art and that claiming such inherent features of computing systems does not run afoul of 35 U.S.C. § 112, first paragraph. The applicant merely seeks to clearly claim a more limited and enumerated set of features that are inherent in the broadest reasonable interpretation of the claimed disclosed subject matter in view of the then current state of the art.

Similarly, in regard to Claim 14, the specification at pg. 7, ln. 5-9 states that the malware can be intercepted from a "network 106" and also when encountered on "distributable media". One of skill in the art at the time of filing would easily have comprehended a "network" to inherently include wired and/or wireless networks (*e.g.*, IEEE 802.11B standard (released 1999) and IEEE 802.11G standard (released 2003) were well known by 30 January 2004, *see* en.wikipedia.org/wiki/IEEE_802.11). Similarly, distributable media would easily be understood by one in the art to include CD-RW, DVD-Rom and/or DVD-RW, especially in like of the specification stating, "floppy disk, flash memory, CD-ROM disk, magnetic tape, **and the like**." *Id.* Where the specification in light of the state of the art at the time of filing is sufficiently clear to support the claimed subject matter, the claims are not deficient under 35 U.S.C. § 112. Applicants' representative respectfully requests that the Examiner withdraw the rejection of claims 10 and 14 under 35 U.S.C. § 112 and pass the application to issue at an early date.

II. Rejection of Claims 1, 4, and 7-24 Under 35 U.S.C. §103(a)

Claims 1, 4, and 7-24 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Lucas et al. (US 6,968,261) in view of Thacker (US Pub. No. 2002/0035696).

As previously stated, applicants' disclosed subject matter relates to detecting malware. The malware evaluator intercepts incoming code/data and searches for malicious code. This can be done by searching the arriving code/data for recognized patterns representative of known malicious code/data. Whereas hackers and the like have come to understand that these searches look for known patterns, the hackers have developed methods of packing malicious executable code to disguise it from traditional virus detecting software. By detecting and unpacking packed code as it arrives, *e.g.*, intercepting it, the Applicants' invention can maintain the advantage over hackers' attempts at propagating malware through packed malicious executables.

Also as previously presented, the invention of Lucas generally relates to searching code for viruses in an "on-access antivirus system" (*see* Lucas, col. 3, line 47). Lucas describes that hackers have determined a particular weakness in anti-virus software that can cause the anti-virus software to "timeout" when system resources become severely taxed (*see* Lucas, col.1, ln. 11-36). Lucas proposes a solution of decompressing compressed files from a hard drive device (Lucas, col. 3, ln. 51-52) on-access and scanning these decompressed files for viruses. In Lucas's proposal, malicious files that have been compressed, so as to bog down a system on

decompression, can be parsed into smaller decompressed pieces to allow for sequential scanning of each decompressed piece for virus signatures by comparison to DATs (Lucas, col. 4, ln. 7-17). Lucas does not discuss intercepting code/data as it arrives at a computer.

Whereas the Examiner has in effect conceded that Lucas does not teach intercepting the code/data as it arrives, as evidenced by the Examiner conducting a further search to find art in support of intercepting code in virus protection systems, the Examiner turns to Thacker to show obviousness.

Applicants' representative disagrees with the Examiner's use of Thacker for support of code interception in a malware protection system that unpacks executables so that they can be analyzed for malicious code. Generally, Thacker describes intercepting code, primarily executables, as they traverse between a network and a computer to prevent them from reaching the computer where they could cause harm (*see* Thacker Figure 1). Thacker further describes the executables caught in the virus trap as being allowed to proceed to the computer after it is determined that the code is safe by "...selecting a by-pass for programs and attachments which are known to be good..." (*see* Thacker at [0013].) However, Thacker teaches away from the invention of the disclosed subject matter in that the invention of Thacker cannot be combined with Lucas in a manner that makes the subject invention obvious without altering the invention of Thacker.

> A prior art reference must be considered in its entirety, *i.e.*, as a whole, including portions that would lead away from the claimed invention. *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 220 USPQ 303 (Fed. Cir. 1983), *cert. denied*, 469 U.S. 851 (1984).
>
> The Federal Circuit has held that teaching away from the art of the subject invention is a *per se* demonstration of lack of prima facie obviousness. *In re Dow Chemical Co.*, 837 F.2d 469, 5 USPQ2d 1529 (Fed. Cir. 1988).
>
> If a reference is cited that requires some modification in order to meet the claimed invention or **requires some modification in order to be properly combined with another reference** and such modification destroys the purpose or function of the invention disclosed in the reference, one of ordinary skill in the art would not have found a reason to make the claimed modification. *In re Gordon*, 733 F.2d 900, 221 USPQ 1125 (Fed. Cir. 1984), emphasis added.

Where Thacker describes the use of the virus trap, the trap is described as operating on executable programs and attachments. This code is not compressed or packed. Further, there is no unpacking of any packed executables described. Moreover, Thacker goes so far as to state that, "Because the virus trap is designed ***to trap executable programs and attachments, it needs no virus detection patterns***, and thus requires no latebreaking virus recognition information from the virus protection industry. The device detects new viruses and therefore is not limited to the viruses which have already been documented in databases." *See* Thacker [0012], emphasis added. Thacker is fundamentally different in that Thacker traps an uncompressed executable code piece and allows it to run in the trap to see if it is malicious before allowing a user to bring it past the trap. This is in direct contrast to Lucas where code that is intentionally and maliciously designed to be huge when unpacked (to bog down a computer and allow a viral attack) is allowed to enter a computer but be unpacked in portions such that each portion can be checked rigorously against known virus signatures and definitions as updated from a virus database (DAT). Thus, Thacker cannot reasonably be combined with Lucas. To do so would require modification of Thacker away from what Thacker has explicitly taught.

Where, Thacker cannot properly be used with Lucas to cure the defect of Lucas with regard to preventing the entry of viruses into the computer, the claims presented are believed to be allowable over the cited art. With regard to Claim 1, independent claim 1 recites, "...a malware evaluator for determining whether incoming data is malware, wherein the incoming data directed to a computing device **is intercepted by the malware evaluator**..." (emphasis added). Contrary to assertions made in the Office Action, the cited references do not disclose or suggest this feature of Applicants' claimed invention. The Applicants' invention explicitly describes intercepting incoming data wherein the invention, "operates on incoming data **as it physically arrives at the computer**...the incoming data does not actually "reach" the computer until it gets past the anti-virus software." (pg. 2, ln, 2-5, emphasis added). The antivirus software of the present application, unlike Thacker, unpacks packed executables and checks them against known viral signatures. Applicants' invention discloses unpacking packed executables for malware detection. Lucas does not teach intercepting malicious code before it reaches a computer, and further, Lucas cannot properly be combined with Thacker to cure this defect.

Further, as previously stated, Applicants' invention contemplates unpacking entire packed executables, without executing the unpacking code included in a potentially malicious

packed executable, to create *corresponding* unpacked executables for analysis. This claimed feature should not be overlooked. The analysis for malware in the subject application is not conducted on an actual unpacked potentially malicious piece of code or data, but rather is conducted on a representation of what the unpacked code/data would look like, but the representation is actually unpacked by a controlled unpacker (*e.g.*, by selectable unpacker modules within the unpacking module) designed to unpack specific file types. This serves to prevent execution of and infection by unpacking of the actual potentially malicious packed executable. This is explicitly claimed in Claim 1, "...receives a packed executable from the malware evaluator and returns an unpacked executable **corresponding** to the packed executable...", (emphasis added). Lucas neither explicitly nor implicitly discloses this feature of the claimed invention. This feature is also missing in any ***arguendo*** proper combination of Lucas and Thacker (which combination applicants' representative asserts remains improper).

The Examiner has rejected Claim 4 based on similar reasoning as applied to Claim 1, Applicants' representative therefore similarly disagrees with the Examiner's position. In particular, independent claim 1 recites, "...**intercepting incoming data** directed to a computing device" (emphasis added). Contrary to assertions made in the Office Action, the cited reference, for the reasons disclosed herein above with respect to Claim 1, does not disclose or suggest this feature of Applicants' claimed invention. Further, Claim 4 states, "...the unpacked executable **corresponding** to the packed executable" (emphasis added). Contrary to assertions made in the Office Action, the cited reference, for the reasons disclosed herein above with respect to Claim 1, does not disclose or suggest this feature of Applicants' claimed invention.

Therefore, based on the above remarks, the Applicants respectfully request that the Examiner withdraw the rejection of Claims 1 and 4 under 35 USC § 102(e) as being anticipated by Lucas.

Claims 7-24 each rely on aspects of unpacking packed executables and thus the cited art suffers the same problems as described above. Applicants' representative therefore further asserts that these claims are patentably distinct over Lucas either alone or in combination with Thacker. For example, Claims 7 and 8 recite features of unpackers employed in returning a corresponding executable; Claims 9 and 10 recite features of intercepting data/code before it reaches the computer; Claims 13 and 15-17 recite features of determining malware with regard to code/data that is not a packed executable; and Claims 19-24 recite features of methods including

corresponding executable aspects, aspects of intercepting code/data from networks and distributable media, and aspects of unpacking entire executables, among others.

CONCLUSION

The present application is believed to be in condition for allowance in view of the above comments and amendments. A prompt action to such end is earnestly solicited.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [MSFTP2193US].

Should the Examiner believe a telephone interview would be helpful to expedite favorable prosecution, the Examiner is invited to contact applicants' undersigned representative at the telephone number below.

Respectfully submitted,

AMIN, TUROCY & CALVIN, LLP

/Himanshu S. Amin/
Himanshu S. Amin
Reg. No. 40,894

AMIN, TUROCY & CALVIN, LLP
24TH Floor, National City Center
1900 E. 9TH Street
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731